

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for monitoring multiple online resources in different formats, the method comprising the steps of:

identifying a plurality of online resources to monitor, the plurality of online resources being stored in a plurality of formats, at least one of the plurality of online resources including data in a non-strict architectural structure;

identifying whether each of the online resources of the plurality of online resources is a non-HyperText Markup Language application;

for each of the plurality of online resources from the non-HyperText Markup Language application, converting the online resource from the non-HyperText Markup Language application to a HyperText Markup Language application;

for each of the online resources of the plurality of online resources, determining whether the online resource meets a minimum level of content structure to allow an Extensible Style Sheet Transform to be used to convert the online resource to the strict formatted file;

converting each of the plurality of online resources that is determined as meeting the minimum level of content structure to a strict formatted file having a common format, wherein the strict formatted file is an Extensible Markup Language application, and wherein data in the plurality of formats of the plurality of online resources is converted into a strict architectural structure which places constraints on a structural location of content identifiers and flags in the respective strict formatted file;

converting each of the plurality of online resources that is determined as not meeting the minimum level of content structure to a strict formatted file, wherein the strict formatted file is a document object model of the online resource;

after converting to the strict formatted file, identifying relevant data in each of the strict formatted files based on the strict architectural structure of the data using an analytic parser; and

comparing the identified relevant data to a most recent archived copy of the identified relevant data to determine whether the identified relevant data has been altered.

2. (Previously Presented) The method of claim 1 wherein at least one of the online resources is a HyperText Markup Language application.

3. (Previously Presented) The method of claim 1 wherein at least one of the online resources is a non-HyperText Markup Language application.

4. (Currently Amended) The method of claim 3 ~~further comprising the step of converting the online resource from the non-HyperText Markup Language application to a HyperText Markup Language application~~, wherein converting the online resource to the strict formatted file comprises converting the HyperText Markup Language application to the strict formatted file.

5. (Previously Presented) The method of claim 1 wherein an Extensible Style Sheet Transform is used to convert each online resource to the strict formatted file.

6. (Canceled)

7. (Previously Presented) The method of claim 1 wherein the strict formatted file is an Extensible HyperText Markup Language application.

8. (Previously Presented) The method of claim 1 wherein each strict formatted file is a document object model of one of the online resources.

9. (Previously Presented) The method of claim 1 wherein the analytic parser is a script that operates on the strict formatted file.

10. (Previously Presented) The method of claim 9 wherein the script identifies relevant data via markers within the strict formatted file.

11. (Canceled)

12. (Previously Presented) The method of claim 1 further comprising the step of storing the identified relevant data within a database.

13. (Previously Presented) The method of claim 1 further comprising the step of automatically notifying a user when the identified relevant data has changed.

14. (Previously Presented) The method of claim 1 further comprising the step of automatically updating a database.

15. (Currently Amended) A system for monitoring multiple files in disparate formats, the system comprising:

a file type identifier module adapted to identify the format of each of a plurality of online resources, at least one of the online resources being in a first format including data in a non-strict architectural structure;

a format conversion module adapted to₂

identify whether each of the online resources of the plurality of online resources is a non-HyperText Markup Language application,

for each of the plurality of online resources from the non-HyperText Markup Language application, convert the online resource from the non-HyperText Markup Language application to a HyperText Markup Language application,

for each of the online resources of the plurality of online resources, determine whether the online resource meets a minimum level of content structure to allow an Extensible Style Sheet Transform to be used to convert the online resource to the strict formatted file,

convert each of the online resources that is determined as meeting the minimum level of content structure to a strict formatted file having a common format, wherein

the strict formatted file is an Extensible Markup Language application, and wherein data in the format of each online resource is converted into a strict architectural structure, and which places constraints on a structural location of content identifiers and flags in the respective strict formatted file;

convert each of the online resources that is determined as not meeting the minimum level of content structure to a strict formatted file, wherein the strict formatted file is a document object model of the online resource;

after conversion to the strict formatted file, an analytic parser adapted to identify relevant data in the strict architectural structure in each strict formatted file;

a resource filter adapted to determine whether the identified relevant data has been altered by comparing the identified relevant data in at least one of the strict formatted files to a most recent archived copy of the identified relevant data.

16. (Previously Presented) The system of claim 15 wherein at least one of the online resources is a HyperText Markup Language application.

17. (Previously Presented) The system of claim 15 wherein at least one of the online resources is a non-HyperText Markup Language application.

18. (Currently Amended) The system of claim 17 ~~further comprising an HTML conversion module adapted to convert the online resource from the non-HyperText Markup Language application to a HyperText Markup Language application~~, wherein the format conversion module is adapted to convert the online resource to the strict formatted file by converting the HyperText Markup Language application to the strict formatted file.

19. (Previously Presented) The system of claim 15 wherein an Extensible Style Sheet Transform is used to convert each online resource to the strict formatted file.

20. (Canceled)

21. (Previously Presented) The system of claim 15 wherein the strict formatted file is an Extensible HyperText Markup Language application.

22. (Previously Presented) The system of claim 15 wherein the strict formatted file is a document object model of the online resource.

23. (Previously Presented) The system of claim 15 wherein the analytic parser is a script that operates on the strict formatted file.

24. (Previously Presented) The system of claim 23 wherein the script identifies relevant data via markers within the strict formatted file.

25. (Canceled)

26. (Previously Presented) The system of claim 15 wherein the identified relevant data is stored within a database.

27. (Previously Presented) The system of claim 15 further comprising a monitoring module adapted to automatically notify a user when the identified relevant data has changed.

28. (Previously Presented) The system of claim 15 further comprising a monitoring module adapted to automatically update a database when the identified relevant data has changed.

29. (Currently Amended) A method for monitoring multiple online resources in different formats, the method comprising the steps of:

identifying a plurality of online resources to monitor, the online resources being stored in a plurality of formats, at least one online resource includes a database and includes data in a non-strict architectural structure;

identifying whether each of the online resources of the plurality of online resources is a non-HyperText Markup Language application;

for each of the plurality of online resources from the non-HyperText Markup Language application, converting the online resource from the non-HyperText Markup Language application to a HyperText Markup Language application;

for each of the online resources of the plurality of online resources, determining whether the online resource meets a minimum level of content structure to allow an Extensible Style Sheet Transform to be used to convert the online resource to the strict formatted file;

converting each online resource that is determined as meeting the minimum level of content structure to a strict formatted file having a common format, wherein the strict formatted file is an Extensible Markup Language application, and wherein data in the online resource in the non-strict architectural structure is converted into a strict architectural structure which places constraints on a structural location of content identifiers and flags in the respective strict formatted file;

after converting to the strict formatted file, identifying relevant data in each of the strict formatted files based on the strict architectural structure using an analytic parser;

remotely updating the relevant data in the database online resource by using a script which updates the relevant data in at least one of the strict formatted files converted from the database online resource and by converting the at least one strict formatted file to an original format of the database online resource.

30. (Currently Amended) A system for monitoring multiple files in disparate formats, the system comprising:

a file type identifier module adapted to identify the format of each of a plurality of online resources having a plurality of formats, at least one online resource including data in a non-strict architectural structure;

a format conversion module adapted to

identify whether each online resource is a non-HyperText Markup Language application,

for each online resource from the non-HyperText Markup Language application, convert the online resource from the non-HyperText Markup Language application to a HyperText Markup Language application,

for each online resource, determine whether the online resource meets a minimum level of content structure to allow an Extensible Style Sheet Transform to be used to convert the online resource to the strict formatted file,

convert each online resource that is determined as meeting the minimum level of content structure to a strict formatted file having a common format, wherein the strict formatted file is an Extensible Markup Language application, and wherein data of the online resource in the non-strict architectural structure is converted into a strict architectural structure, which places constraints on a structural location of content identifiers and flags in the respective strict formatted file;

convert each online resource that is determined as not meeting the minimum level of content structure to a strict formatted file, wherein the strict formatted file is a document object model of the online resource;

after conversion to the strict formatted file, an analytic parser adapted to identify relevant data in each strict architectural structure in the strict formatted file; and
a resource updater to update the identified relevant data in a database.

31. (Previously Presented) The method of claim 1, wherein identifying relevant data in the strict formatted file comprises identifying data flags or identifiers in the strict architectural structure to identify the relevant data.

32. (Previously Presented) The system of claim 15, wherein the analytic parser is adapted to identify data flags or identifiers in the strict architectural structure to identify the relevant data.

33. (Previously Presented) The method of claim 29, wherein identifying relevant data in the strict formatted file comprises identifying data flags or identifiers in the strict architectural structure to identify the relevant data.

34. (Previously Presented) The system of claim 30, wherein the analytic parser is adapted to identify data flags or identifiers in the strict architectural structure to identify the relevant data.

35. (Currently Amended) A method for monitoring multiple online resources in different formats, the method comprising the steps of:

identifying a plurality of online resources to monitor, at least one resource of the plurality of online resources being stored in a first format including data in a non-strict architectural structure;

identifying whether each of the plurality of online resources is a non-HyperText Markup Language application;

for each of the plurality of online resources from the non-HyperText Markup Language application, converting the online resource from the non-HyperText Markup Language application to a HyperText Markup Language application;

for each of the plurality of online resources, determining whether the online resource meets a minimum level of content structure to allow an Extensible Style Sheet Transform to be used to convert the online resource to the strict formatted file;

converting each of the plurality of online resources that is determined as meeting the minimum level of content structure to a strict formatted file, wherein the strict formatted file is an Extensible Markup Language application, and wherein data in the first format of the at least one online resource is converted into a strict architectural structure ~~which places constraints on a structural location of content identifiers and flags in the respective strict formatted file;~~

converting each of the plurality of online resources that is determined as not meeting the minimum level of content structure to a strict formatted file, wherein the strict formatted file is a document object model of the online resource;

after converting to the strict formatted file, identifying relevant data in the strict formatted file based on the strict architectural structure of the data using an analytic parser;

comparing the identified relevant data to a most recent archived copy of the identified relevant data to determine whether the identified relevant data has been altered; and

automatically updating altered identified relevant data to a new archived copy.